Intelligent Systems, Control and Automation:
Science and Engineering

G. Labinaz
M. Guay

# Viability of Hybrid Systems

## A Controllability Operator Approach

Springer

# Viability of Hybrid Systems

International Series on
INTELLIGENT SYSTEMS, CONTROL, AND AUTOMATION:
SCIENCE AND ENGINEERING

VOLUME 55



For further volumes:
www.springer.com/series/6259

G. Labinaz • M. Guay

# Viability of Hybrid Systems

## A Controllability Operator Approach

Springer

G. Labinaz
Queen's University
Department of Chemical Engineering
K7L 3N6 Kingston, Ontario
Canada
glabinaz@gmail.com

M. Guay
Queen's University
Department of Chemical Engineering
K7L 3N6 Kingston, Ontario
Canada
martin.guay@chee.queensu.ca

ISBN 978-94-007-2520-1 e-ISBN 978-94-007-2521-8
DOI 10.1007/978-94-007-2521-8
Springer Dordrecht Heidelberg London New York

Library of Congress Control Number: 2011939264



*Cover design*: VTeX UAB, Lithuania

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

# Acknowledgements

We would like to express our appreciation to Karen Rudie and Mohamed Bayoumi for their support and assistance. We would also like to express our gratitude to Pierre Belanger for getting GL started in research.

We would also like to express our gratitude to the Department of Chemical Engineering at Queen's University for their continued support.

Much of this work is based on the Phd thesis of GL during which time he would like to gratefully acknowledge the assistance and friendship of fellow graduate students, in particular, Youcef Derbal, Peter Gallant, Ahmad Masoud and Laurie Ricker. Also, GL acknowledges the support and friendship of Steve and Launi Fotopoulos, Karina McInnis, Marino and Debbie Labinaz and Tom Sansom. Lastly, this work could not have been completed without the support of GL's parents Amalia and Emilio. It is to them that this book is dedicated.

MG dedicates this book to Mélanie, Samuel and Bonnie. Thank you for your love and support.

# Contents

# 1

# Introduction

*Hybrid Systems: Why, How?*

This introductory chapter provides an overview of the problems addressed in this book, and a summary of the book and its contributions.

The chapter is organized as follows. Section 1.1 provides a brief history of hybrid systems and motivates this work and its approach. A summary of the book and its organization are given in Section 1.2 and Section 1.3 provides some concluding remarks.

## 1.1 Motivation and History

The field of Hybrid Systems (HS) is in the midst of rapid changes, evolution and development. One generally accepted attribute of hybrid systems is that they include both discrete (or digital) and continuous behaviour. By 'include', we mean that both discrete and continuous variables, dynamics, and, conditions, which we will refer to collectively as 'domains', are required in order to fully characterize the behaviour of interest.

With a term like 'hybrid' modifying another term like 'system', the need to precisely describe the term 'hybrid system' is important. The need for and difficulty in assigning a precise meaning to the term hybrid system is compounded by the fact that ambiguity in what hybrid systems are has allowed a number of different problems and areas to come to be referred to as being 'hybrid'. In this section, we plan to describe the meaning of hybrid systems,

G. Labinaz, M. Guay, *Viability of Hybrid Systems*, Intelligent Systems, Control and Automation: Science and Engineering 55,
DOI 10.1007/978-94-007-2521-8_1,

to place them in context and to motivate the need to study hybrid systems and the approaches taken in this work.

The problems, questions and methods that arise in hybrid systems would seem to be distinct enough from other existing areas of systems and control, that we feel justified in referring to hybrid systems as a field of systems and control. To the best of our knowledge, a generally accepted definition of hybrid systems does not exist and we do not attempt to give one here. In fact, at this fairly early stage in the field's development when there still are many open problems and diverse approaches being explored, it is reasonable to leave hybrid systems undefined. Instead, it does seem possible to characterize hybrid systems according to attributes of the problems and/or methods used in solving these problems in a definitive way. It would seem that diverse yet related investigations will help crystallize a scientific consensus based on fundamental needs and ideas of hybrid systems.

A question may arise "wasn't this behaviour present earlier"; to which one would answer "yes". However, it is the recognition that the inclusion and integration of continuous and discrete behaviour into the modeling formalism better reflects reality, that problems can be posed and answered combining these two domains, and that solving these problems is both theoretically interesting and practically valuable. Furthermore, hybrid–type behaviour also arises by allowing a larger class of possible control action. Hybrid systems in terms of physical systems, have existed for decades. However, the formalization of the problems and methods that they involve is much more recent.

A brief recent history of the hybrid systems field is given next, with a focus towards control of hybrid systems. This will be done without careful reference to specific literature but rather relative to main contributions and their apparent impact on the field. A more formal and specific literature review is given in Section 2.4. It is a recent history that is in the stages of significant and rapid development and evolution.

In the early 1980's, the work of Ramadge and Wonham [78] on control of supervisory control of discrete–event systems introduced a control–theoretic approach to problems involving logical plant descriptions and performance specifications. This initiated the field of Discrete–Event Systems (DES) and spawned a rich array of investigations into problems of control, observation, decentralization and hierarchy, among others. In the late 1980's, Varaiya and colleagues [43] and Benveniste and colleagues [16] introduced and examined questions involving systems having continuous and discrete domains. The work of Varaiya and colleagues was motivated by problem in highway automation [91]. The coupling between practical and theoretical problems is one that was present in the early stages of hybrid systems and continues to be the case. Benveniste's approach to hybrid systems was to characterize a system as hybrid based on a computer programming language and the behaviours that it could describe. Also in the late 1980's, work by Caines and colleagues [23] initiated an approach for integrating logic and control, or as put in their early work, "...to show that Artificial intelligence and systems and control theory have an intersection (or product!) containing a set of problems that possess the conceptual features of both subjects." [23]. Also in the late 1980's a Task Force report on future trends and directions for the control systems field was published [63]. Many of the comments and recommendations in this report reflect problems and approaches that are being investigated as part of the hybrid system activities.

In the early 1990's, an annual Hybrid System Workshops series was initiated. The first was held at the Mathematical Sciences Institute, Cornell University in 1991. The Workshops address a variety of hybrid systems questions related to modeling, control, analysis, verification, design, simulation and applications, with the post–Workshop proceedings being published in the *Lecture Notes in Computer Science* series. The diversity of methods and views towards hybrid systems becomes apparent in these collections. As has often been noted and remarked by hybrid researchers, hybrid systems are being studied by persons from a variety of disciplines including engineering, mathematics, and computer science. This fact could be said to be seen as further contributing to the diversity of interests, methods, and 'tools' found within the hybrid systems field.

In the early 1990's, Kohn discussed an approach to intelligent real–time control based on a *Declarative Control Architecture* [47]. A high-level decomposition of system behaviour was given based on the following three principles: *conservation*, *constraint*, and *invariance*. These apply to both physical and non–physical systems and can be used in guiding the problem definitions and the required theoretical techniques.

In the first Lecture Notes series published in 1993, the work of Nerode and Kohn [68] considered a number of issues, problems and approaches for hybrid systems, in particular, their control, and has provided direction for a number of different investigations. An underlying premise of their work is the treatment of the plant and controller of the hybrid system as an automaton. On the other hand, the work of Branicky, appearing in full in [18], of considering hybrid systems relative to the continuous domain is a complementary approach. Both of these approaches and much of the work by others on control of hybrid systems, consider questions of optimality and/or state invariance in the presence of hybrid phenomena.

Although there is no 'explicit' consensus on what makes a system hybrid, we could suggest that there would seem to be some 'implicit' consensus developing. A list of some observations that could be taken as points that are beginning to form a consensus is as follows: (i) HS involves continuous and discrete domains, (ii) a main issue in HS is being able to consider continuous and discrete domains in a unified and consistent way, (iii) there is a rich collection of existing hybrid models describing a number of hybrid phenomena, (iv) solutions to control problems will often be algorithmic and require approximations, (v) there is a rich set of practical problems, both small and large–scale, to which HS can be applied. We remark about Item (ii) that although this appears to be a fairly well–accepted assumption and perhaps even 'intuitively obvious', the clear advantages, gains, and needs to consider the two domains in a unified and consistent way has not been formally proven to the best of our knowledge. Of course, the fact that these two domains appear in many situations and applications could be taken as a strong motivation and justification for this need.

In [48], the following problem is stated as the fundamental problem of hybrid systems:

> *Find algorithms which, given continuous plant differential equations and plant performance specifications (which may include logical constraints), extract digital control programs (mode switching programs) that force the state trajectories of the system to obey their performance specifications.*

A typical hybrid closed-loop system is shown in Figure 1.1. The *plant* represents the process to be controlled. The *Analog-to-Digital (AD) Converter* maps continuous measurements into symbolic inputs that are supplied to the *Digital Control Automaton (DCA)*. The DCA control strategy operates on a (finite) number of inputs to generate a (finite) number of outputs. These outputs are sent to the *Digital-to-Analog Converter (DA)* which generates a continuous control input based on the control automaton's output which can then be applied to the plant. The AD and DA are often grouped together and referred to as the *interface*.

As discussed in [70], two approaches that can be taken when considering systems and their control, is what is referred to as 'bottom-up' versus 'top-down'. In the context of control of hybrid systems, the former would begin, for example, by partitioning the state space and designing a control solution to satisfy the performance specifications while the latter would use the performance specifications to derive the required partition. We believe that this distinction is fundamental to hybrid systems and to the problems that can be solved. We adopt the 'top-down' philosophy in this work.

Taking a 'top-down' approach has implications at all problem stages; we consider here those imposed on the hybrid model. What we need to ask (and answer) is "what should be the basis for how the way of decomposing the continuous space(s)"? The answer we give, guided by the 'top-down' methodology, is that the decomposition should be based on the available control laws. Any uncon-

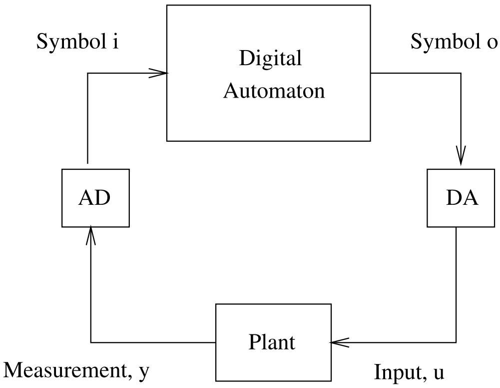


**Fig. 1.1.** Typical hybrid closed-loop system (from (Branicky *et al.*, 1994)).

trolled transition behaviour should be able to be accounted for by the description of the plant. Along with being consistent with the top-down approach, this point of view also avoids the need to consider questions arising, for example, in the incorporation of logical reasoning as part of the control decision capabilities. We believe that having logical reasoning capabilities within the hybrid setting is important in achieving a unified hybrid framework. However, this has not been considered in this work. A logical framework would seem to need to be defined in a consistent and complementary way to the requirements imposed by the continuous–time qualitative properties.

We next turn to a more detailed discussion on the motivation of this work. The main issue that we consider is to ensure that certain qualitative properties of the dynamical systems exhibiting hybrid behaviour can be ensured through control. The control problem requires the design of control devices to generate control actions that realize these properties.

There are two main types of hybrid phenomena that we are interested in: (1) transitions in states, and (2) transitions in dynamics. The first is not explicitly modeled but rather is accounted for as part of the control design process. The second is explicitly modeled. A modeling framework is considered that accounts for (1) multi–valued state evolution, and (2) nonsmooth state constraint performance specifications. The first phenomena is considered as a means of dealing with multiple dynamic modes and model uncertainties. The second phenomena provides a means to handle encoding of logical conditions into the state space.

Next the viability control problem is stated where the desired control action is required to keep states starting within some user specified set to remain within this set for all time.

*Viability Control Problem (VCP): Given the dynamics describing the evolution of the system state over time, the set of possible control action, and the performance specifications in the form of constraints that the systems state must satisfy for all time, find a control decision methodology that selects from the available control action to ensure that the system state satisfies the performance specifications.*

Referring back to the fundamental problem of hybrid systems, the above problem is subsumed by the more general fundamental problem statement. We include 'control' explicitly in the above descriptions of the problem to make clear that it intrinsically requires that a control solution be found. Omitting the term 'control' defines the same qualitative problems except that a control solution is not required to be found. This can be then considered as a problem of analysis (or verification) versus control (or synthesis). In Figure 1.2, we illustrate the VCP problem for continuous–time system dynamics with discrete phenomena, where $VS$ denotes the state constraint set required for VCP.

## 1.2 Summary and Organization

Throughout this book, we assume that (i) the system dynamics are given by time-invariant (autonomous), finite–dimensional systems (ordinary differential equations or ordinary differential inclusions), (ii) all continuous plant states can be observed with observations

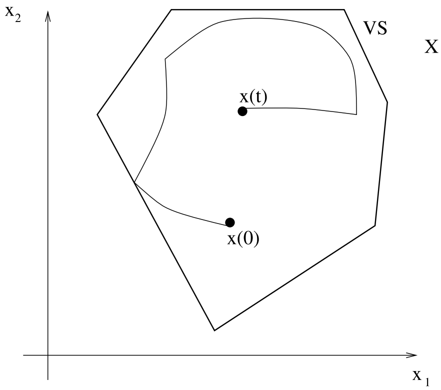


**Fig. 1.2.** Illustration of the Viability Control Problem.

---

taken at uniform time intervals separated by $\Delta > 0$, and (iii) disturbances on the plant are ignored.

The following give chapter summaries for the body of the book.

CHAPTER 3 HYBRID MODEL

A modeling formalism is adopted in which a continuous–time plant, described by an ordinary differential equation (or inclusion) is coupled to an control automaton through AD and DA mappings. The plant and control automata can be viewed as input–output devices. We refer to this as an instance of a simple hybrid system. This model captures the five characteristics of controlled hybrid systems that we are interested in. Three forms of uncertainty, transition dynamics, structural uncertainty, and parametric uncertainty are introduced and a way to express each in a manner that is consistent with the HCLS is given. Transition dynamics are considered in detail, with four transition dynamics models given. These models can be considered to vary from having 'minimal' knowledge to 'maximal' knowledge of the transition behaviour. We examine the basic three–tank example using this modeling and consider a variety of modifications that can be made to the basic problem. This problem is examined in some detail at this stage since it is used throughout the remainder of the book as an example problem. We will not make further specific mention of this example relative to the summaries in Chapters 3–4 below.

Two related notions of a hybrid trajectory are given based on the instance of the simple hybrid system model. A means of ordering segments of these trajectories is provided. A general relationship that applies to each of the three control law classes is established between continuity of a mapping, the existence of a fixed point of this mapping, and the existence of a hybrid trajectory that satisfies the qualitative property defined by the fixed point of the mapping.

Chapter 4 Viability

Control design for ensuring viability in the case of time–independent and time–dependent state constraints is considered. In both cases, the control laws are assumed to be generated by a finite control automaton. Three classes of admissible control law sets are considered: (i) piecewise constant control ($PWC^{\Delta}$), in which a transition between constituent control systems is allowed only at the sampling instants, (ii) piecewise constant control with finite switching ($PWC^{\Delta,k}$), in which a finite number of transitions between constituent control systems is allowed within the sampling interval, and (iii) piecewise constant with polynomial control ($PWCPC^{\Delta,k}$), in which the control law for the sample interval is determined at each sampling instant, with the control law choosing the constituent system (as in the case of sample switching) as well as the continuous control law to be applied for the sampling interval taken as a polynomial of fixed order in time.

For the time–independent viability problem, an approach is given using the solution set. Piecewise constant control comes equipped with a finite set of control laws, this not being the case for the other two control law classes. One approach to dealing with this non–finiteness is by extending the approach in the finite case. An analogous theoretical basis for the time–dependent problem is given.

Chapter 5 Robust Viability

Three forms of uncertainty for hybrid systems are considered: transition dynamics, structural uncertainty, and parametric uncertainty. Two extensions to the controllability operator are introduced for handling uncertainty of hybrid systems: the uncertain controllability operator and the uncertainty operator. The uncertain controllability operator encodes the effect of uncertainty directly into the computation of the controllability operator. The uncertainty operator encodes the effect of uncertainty onto the nominal value of the controllability operator.

Chapter 6 Viability in Practice

Two simulation applications of viability are considered. The first is an Active Magnetic Bearing system in which viability is satisfied by computation of the reachable set for a differential inclusion. The second application is that of a batch polymerization process in which viability is satisfied by cascade control of a viable controller with an existing PID controller. In both cases, satisfaction of viability is demonstrated through simulation.

Chapter 7 An Operator Approach to Viable Attainability of Hybrid Systems

The problem of viable attainability is addressed based on the operator approach initiated by Nerode and colleagues for viability of hybrid systems. Firstly, attainability is addressed whereby attainability refers to reaching some target set of state space within some finite time horizon. This is done by introducing an attainability operator and providing an algorithm for computation of the attainability kernel. Having specified attainability, viable attainability whereby both viability and attainability are required to be satisfied is achieved by intersection of the controllability and attainability operators. Attainability and viable attainability are demonstrated using the three fluid–filled tank example.

Chapter 8 Some Topics Related to the Controllability Operator

In this chapter, we collect facts and properties of the controllability operator. Firstly, we show that the $\varepsilon_n(x_0)$–balls which are removed as part of the satisfaction of viability are continuous functions. This leads to establishing continuity of the controllability operator. Secondly, we consider the lattice properties of the control laws. Two orderings of the control law classes are defined, one weak and one strong ordering. Having this, it is established that the set of control law classes with the order relation and over set intersection and union form a lattice. Next conditions for satisfying the order relations are derived. Thirdly, homotopies are defined to consider the variation in the value of the controllability operator relative to the base admissible control law class $\mathrm{PWC}^{\Delta}$ which corresponds to the collection of piecewise continuous functions over the sampling interval $\Delta$.

## 1.3 Summary

The term *viability* appears to have its origin in continuous–time Viability Theory by Aubin and colleagues [11], [7]. On the other hand, the term *invariance* has been adopted in the work of Clarke and colleagues [27], [28] within the context of nonsmooth analysis, with weak invariance being an equivalent notion to viability. In this work, we choose to adopt the term *viability* mainly since it has come to be used within the hybrid systems field.

# 2

# Literature Review

*Hybrid Systems: Who, What, When?*

In this chapter, we review existing literature on hybrid systems that is related to this work. This is carried out by first considering three specific approaches to viability of hybrid systems, these being due to Nerode and colleagues, Aubin and colleagues and Deshpande–Varaiya. This review is carried out in Sections 2.1–Section 2.3. In Section 2.4, literature related specifically to Chapters 3–8 is then reviewed. Some concluding remarks are made in Section 2.5. Given that hybrid systems is a field that encompasses a variety of problem domains and disciplines, there exists a vast body of related work. Although we do provide some general overview of the field, we will focus on approaches to hybrid systems, and in particular their control, that have the most direct impact on this work.

## 2.1 Nerode *et al* Approach to Viability of Hybrid Systems [49], [70]

In [49], the fundamental problem which is tackled is to extract, given a continuous plant simulation model and performance specification on plant state trajectories, a finite automaton which forces the hybrid system to satisfy the performance specification. More specifically, they wish to capture conditions under which finite state control automata exist which ensure viability is satisfied. The situation is such that Kohn *et al* assume that they are given

G. Labinaz, M. Guay, *Viability of Hybrid Systems*, Intelligent Systems, Control and Automation: Science and Engineering 55,
DOI 10.1007/978-94-007-2521-8_2,

a continuous feedback control function which enforces viable trajectories for a plant and that the objective is to investigate how to extract finite automaton which exhibit controllable and observable behaviour and enforce the same viability as in the continuous time case.

In this work, all hybrid systems are assumed to be simple hybrid systems with fixed control intervals. A simple hybrid system runs open loop within a time interval $[n\Delta, (n+1)\Delta]$ based on a control function $c_n$ and disturbance $d_n$ supplied at time $n\Delta$. The control automaton receives as input at time $n\Delta$ the current state $x$ of the plant, then runs open loop with no further inputs until the time $(n+1)\Delta$. Based on its state at time $(n+1)\Delta$, the control automaton transmits a new control function $c_{n+1}$ to the plant to be used for the time interval $[(n+1)\Delta, (n+2)\Delta]$ and this process repeats. It is assumed that the plant is described by a vector first order differential equation

$$\dot{x} = f(x, c, d)$$

with parameters $c, d$. It is further assumed that

$$\dot{x}(t) = f(x, \tilde{c}(t), \tilde{d}(t))$$

is such that for any time $t_0$, for any initial state $x(t_0)$, for any admissible control and disturbance functions $\tilde{c}(t), \tilde{d}(t)$ defined on $[0, \infty]$, there is a unique solution $x(t)$ defined on $[0, \infty]$ satisfying the differential equation.

The initial value of $\dot{x}(t)$ for the interval $[n\Delta, (n+1)\Delta]$ is not inherited from the previous interval, but is computed from the differential equation based on the current plant state, the initial value of the new control function and the new disturbance at $n\Delta$. This results in the vector field changing direction abruptly at time $n\Delta$ which is characteristic of hybrid control.

Since the plant differential equation is taken as autonomous in this work, the behaviour in an interval of length $\Delta$ is the translate of the behaviour in any other interval of length $\Delta$. Therefore, $c(t)$ and $d(t)$ will be assumed to be defined on $[0, \Delta]$ and translate them by $n\Delta$ for use on the interval $[n\Delta, (n+1)\Delta]$.

**Definition 2.1.** *The continuous plant induces an automaton, which is called the $\Delta$-*plant automaton *associated with a simple hybrid system. It has two input alphabets, the set $D$ of admissible disturbance functions and the set $C$ of admissible control functions. Its set of internal states is the set of plant states. Its transition function assigns to input letters control $c(t)$ and disturbance $d(t)$ and automaton current state $s_0$, the new automaton state $x(\Delta)$ where $x(t)$ is a plant state trajectory such that $x(0) = s_0$ is the solution to the differential equation*

$$\dot{x} = f(x(t), c(t), d(t)).$$

The viability set is denoted as $VS$, a subset of plant states which is usually assumed to be closed and compact.

**Definition 2.2.** *A trajectory $x(t)$ over an interval of time $[0, \Delta]$ is called* viable *of for all $t$ in that interval, $x(t) \in VS$. Similarly, a trajectory extending over $[0, \infty]$ is* viable *if for all $n \geq 0$, the trajectory $x_n(t) = x(t - n\Delta)$ over $[0, \Delta]$ is viable.*

Associated with viability of a simple hybrid system are three definitions of local graphs given next.

**Definition 2.3.** *The* abstract viability graph *is an obvious analogue to the viability kernels of continuous time systems. Nonempty closed compact subsets of this graph and closed viability sets lead to finite automata that enforce viability.*

**Definition 2.4.** *The* sturdy local viability graph *is such that nonempty closed compact subsets lead to finite automata that force viability and also are "safe" under small errors in state and control measurements.*

**Definition 2.5.** *The $e$-*sturdy local viability graph *represents those hybrid systems with a sensor of plant states with error bounded by a fixed $e$. This leads to finite state control automata whose analog to digital converter, or sensor of plant state has error bounded by $e$ and also enforces viability.*

The latter two graphs are not developed extensively in [49] and so will not be considered here in any detail. Next, definitions for nodes and edges of the abstract local viability graph are given.

**Definition 2.6.** *The nodes of the local viability graph are those pairs*

$$(c_0, s_0) \in C \times VS$$

*such that for any disturbance $d_0 \in D$, the trajectory $x(t)$ determined by $d_0$, control $c_0$ and initial state $s_0$ is viable.*

**Definition 2.7.** *There is a directed edge from node $(c_0, s_0)$ to node $(c_1, s_1)$ if and only if*

1. *$(c_0, s_0)$ and $(c_1, s_1)$ both nodes of the local viability graph and*
2. *There is a disturbance $d_0 \in D$ such that the trajectory $x(t)$ with disturbance $d_0$ and control $c_0$ and initial condition $x(0) = s_0$ has $x(\Delta) = s_1$.*

The pair $(c_0, s_0)$ is referred to as the *tail* and $(c_1, s_1)$ is referred to as the *head* of the directed edge. There may be nodes of the abstract local viability graph that are not heads nor tails of any edges. In this case, these nodes are dropped at the beginning of the construction of the graph.

For the local viability automaton

- The input alphabet is the set of viable plant states $VS$.
- The states are the set of controls.
- The non–deterministic transition relation maps a pair $(c_0, s_1) \in C \times VS$ to a control $c_1$ if and only if there exists an edge in the abstract local viability graph with tail $(c_0, s_0)$ and head $(c_1, s_1)$. This is a partially defined transition relation. The interpretation is that $c_0$ should be thought of as the control used in the previous control interval which has, due to a disturbance, produced the current plant state $s_1$. Then, with input letter $s_1$ when in local viability automaton state $c_0$, the local viability automaton moves to state $c_1$ and outputs letter $c_1$.

Next we consider viability over the interval $[0, \infty]$. Assume we are given a directed graph $T$ which consists of a non-empty set $T$ of nodes and a subset $E$ of $T \times T$ of its directed edges such that each node is incident on at least one edge. Each subset $T'$ of $T$ defines a subgraph with edges $E' = E \cap (T' \times T')$. A path is a finite or infinite sequence of edges such that the head of each edge is the tail of the next edge. An end node of a graph is a node which is not the tail of any edge in that graph. Let $P(T)$ denote the power set of $T$.

**Definition 2.8.** *Suppose graph $T$ is given.*

1. *Define a monotone decreasing operator $F : P(T) \to P(T)$ by letting $F(T')$ be the set of nodes of $T'$ which are not end nodes of $T'$ and which are on at least one edge of $T'$.*
2. *For each ordinal $\alpha$, define an operator $F^\alpha : P(T) \to P(T)$ by transfinite induction as*
   *a) $F^0(T') = F(T')$,*
   *b) $F^{\alpha+1}(T') = F(F^\alpha(T))$,*
   *c) $F^\lambda(T') = \bigcap_{\alpha > \lambda} F^\alpha(T')$ if $\lambda$ is a limit ordinal.*

**Proposition 2.9.** *Suppose that $T' \subseteq T$.*

1. *Then $T'$ is a fixed point of $F$ if and only if every node of $T'$ is the initial node of some infinite path in $T'$.*
2. *There is a least ordinal $\alpha$ such that*
$$F^{\alpha+1}(T') = F^\alpha(T)$$
3. *If $\alpha$is the least ordinal such that $F^{\alpha+1}(T') = F^\alpha(T)$, then $F^\alpha(T')$ is the largest fixed point of $F$ contained in $T'$.*

**Proposition 2.10.** *Suppose that:*

1. *The nodes of $T$ are elements of a separable metric space and*
2. *$T'$ is a subgraph of $T$ such that for every ordinal $\alpha$ and for every end node of $F^\alpha(T')$ or node on no edge of $F^\alpha(T')$, there is a neighbourhood containing that node and no other node of $F^\alpha(T')$. (Here we interpret $F^0$ to be the identity map on $P(T)$.)*

*Then*

1. *The least ordinal $\alpha$ such that $F^\alpha(T') = F^{\alpha+1}(T')$ is a countable ordinal.*
2. *If $T'$ is closed, then $F(T')$ is closed.*